Chapter 17

PHY-LAYER OF 802.16 MOBILE WIRELESS ON A HARDWARE ACCELERATED SOC

Stylianos Perissakis[1], Frank Ieromnimon[1], Nikolaos S. Voros[2]
[1]*Intracom Telecom Solutions S.A.*
[2]*Technological Educational Institute of Mesolonghi, Dept. of Telecommunications Systems & Networks (consultant to Intracom Telecom Solutions S.A.)*

Abstract: Modern-day wireless systems are characterized by short development windows, coupled with a large volume of requirements that are often antagonistic to each other, such as ever-increasing processing power, placed against the need to keep power consumption from growing, and also the need to improve on the efficiency of use of the RF spectrum. Traditional ASIC design approaches are severely strained by these requirements and alternatives are eagerly searched for in many disciplines. One promising alternative is the use of reconfigurable ASIPs, which offer the promise of speedy development and upgrade of complex systems, combined with the ease of use and relative safety of software-programmable components. MORPHEUS is one such platform that is evaluated in this case study by means of a portion of the PHY-layer for the popular 802.16e/j wireless standards.

Key words: Wireless standards, ASIPS, programmability.

1. TARGET SYSTEM OVERVIEW

The application targeted by Intracom Telecom Solutions is part of the emerging IEEE 802.16j standard. The latest standard currently in force from the IEEE 802.16 family [1] is 802.16e, the basis for Mobile WiMAX technology [2]. This standard mandates the use of Orthogonal Frequency Division Multiple Access (OFDMA) technology for the physical layer [3] and provides all necessary support in the physical and MAC layers for mobility management, such as network entry, handover, etc. The next standard, 802.16j [4], currently in preparation, extends the concepts defined in 16e by adding the possibility of multi-hop communication between

mobile and base station. For this, the Relay Station entity is defined. The Relay Station is connected to the base station on one side and to a group of mobile stations on the other. The connection to the base station, where the relay acts more or less as a subscriber/mobile station, is called the “relay link”, while the connection to the mobiles, where the relay acts as a simple base station, is called the “access link”. BS-MS communication now may take place over two hops (BS-RS and RS-MS), which can be advantageous because a poor channel is divided into two better ones, allowing for more spectrally efficient modulation and coding schemes. Alternatively, the range of a network cell can be extended, with relays placed near its periphery, serving distant mobiles. The 802.16j standard will reuse the OFDMA physical layer from. 16e, with some minor enhancements regarding mainly the frame structure, and will make significant amendments to the MAC layer.

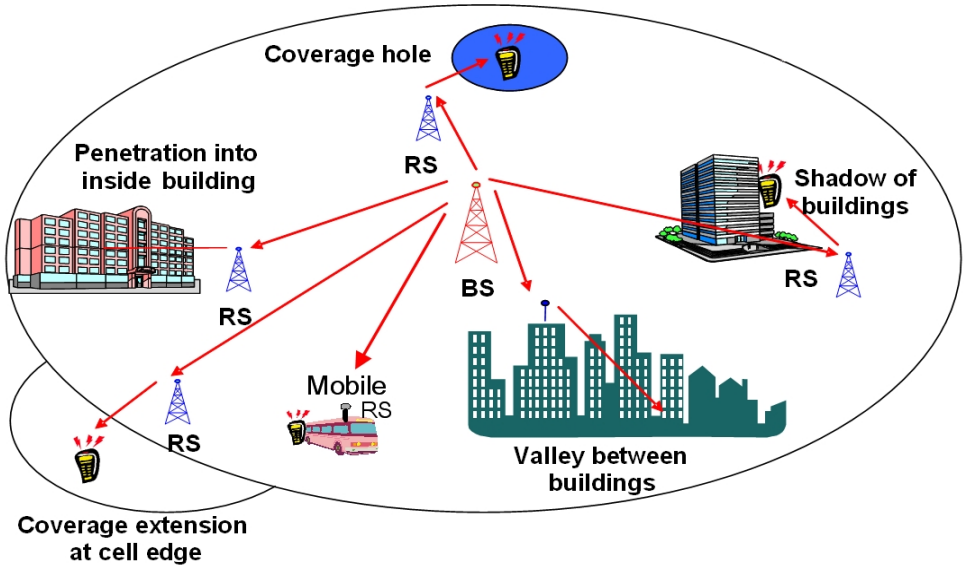

Figure 17-7. Multihop network concept

Relays can be fixed (located e.g. on rooftops, lamp posts, etc), nomadic (transportable, e.g. on trucks) or mobile (on buses, trains, etc). Here we focus on the case of a Mobile Relay Station (MRS). Figure 17-1 demonstrates the concept of a multihop network, including an MRS mounted on a bus that provides service to passengers onboard. As the MRS moves within an area, it will have to perform handover between different base

stations (when crossing from one network cell to another). At the same time, the group of mobile stations it supports will also change dynamically over time. The physical layer mode used in each cell is determined by the base station that serves it. As the propagation environment differs from cell to cell (e.g. urban, suburban, rural), different base stations may require different physical layer modes. While simple terminals, supporting only the mandatory modes, are still backward compatible with all base stations, they need to be able to support the advanced modes in order to take advantage of them. The same holds for a MRS, which acts as a terminal on the relay link.

The use of a reconfigurable platform for this application is attractive, since it offers:

- Long-term flexibility (the ability to adapt to future amendments of the standard)
- Short-term flexibility (the ability to switch transmission modes dynamically, e.g. during handover). Note that a limited initial selection of transmission modes can be extended in the field with additional modes, so it is desirable not to commit to any specific algorithms at system design time, as would be the case with an ASIC platform.

For the first trial of the MORPHEUS concept, we focused on aspects such as exploitation of available concurrency, as well as evaluation of the efficiency of the algorithm mapping process, i.e. how well do the MORPEHUS HREs accommodate the chosen application components.

1.1 Specification for the Envisaged Application

Taking into account the final position on the availability and capacity of the MORPHEUS embedded blocks, Intracom Telecom Solutions performed a study so as to select functional blocks that would a) fit in these embedded blocks and b) match the processing nature of these blocks. The result of this line of reasoning was the decision to use the FlexEOS and DREAM HREs described in Chapter 4 and Chapter 5 respectively.

A computationally intensive word-level processing block was selected (FFT) for the DREAM section (an 128-point FFT), while a QAM symbol demapper was selected for the FlexEOS section. Therefore the selected functionality focuses on that part of the physical layer of IEEE 802.16j standard that would meet both the above capacity constraints and the processing nature of the embedded blocks.

Additional Mobile Station Receiver blocks are also implemented in MORPHEUS chip, such as Cyclic Prefix Removal and Guard Removal. These blocks are implemented by employing data transfers of specific

profiles, involving internal memory. The part of the receiver chain that will be implemented is shown in Figure 17-2, while the blocks included in the chain are briefly described as follows:

- Data Input: Continuous stream of complex time-domain data. Logically the stream consists of *Frames*, each frame consisting of a number of *Symbols*, each symbol consisting of 144 complex samples. Depending on frame configuration, we have 1 preamble symbol, a number of downlink data symbols (typically 28), and a number of "garbage" data samples, corresponding to the uplink period.

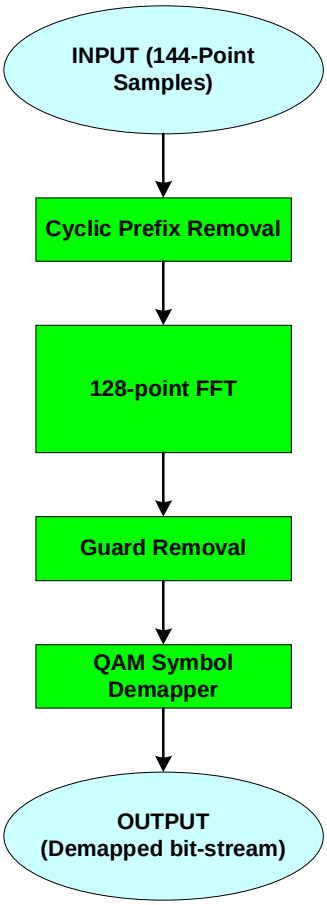

Figure 17-8. Outline of the proposed wireless application example

- Cyclic Prefix Removal: Based on control information coming from the Preamble Detection/Synchronisation Block, a number of data

samples from the continuous input stream are discarded, and a continuous 128-sample window of data (or complex time-domain symbol)_is passed to the FFT downstream..

- FFT: Performs the Fourier Transform on every group of 128 samples, resulting in a complex frequency-domain signal, with 128 points for every input symbol.
- Guard Removal: For normal operation this block removes the guard subcarriers from each symbol.
- QAM Symbol Demapper: The complex frequency-domain data are points in the phase-amplitude plane that represent groups of data bits, the number of which depends on the chosen modulation mode, e.g. 2 bits for QAM4, 4 bits for QAM16, etc. The QAM Demapper converts the points of the complex plane to the groups of data bits that were originally mapped into the constellation of points and converted into a time-domain symbol by the transmitting IFFT.

2. IMPLEMENTATION CONSIDERATIONS

As already outlined briefly, the mapping of the various components of Figure 17-2 was driven by estimates on the capacity of each HRE to handle the computational load. Therefore, the much-larger capacity DREAM array of reconfigurable arithmetic elements was best suited for the FFT block. However, the required buffer space for storing working data was selected to be inside the MORPHEUS chip Local SRAM. The SRAM size of 256 Kbytes is enough to cover double-buffering of FFT sample-data. This feature allows speed optimization, as it allows concurrency between data processing by the DREAM ALUs and data transfers between local SRAM buffers and the external memory that represents the MORPHEUS chip environment (i.e. data source & sink).

The task of QAM demapping was delegated to the FlexEOS array. The much smaller size of that particular HRE is a severe constraint that has to be met by the application design. Thus, rather than be prolific with the description of the structures to be rendered in VHDL, we rely again on the fast memory-to memory transfer capabilities furnished by the MORPHEUS chip, to offload some of the “housekeeping” computation onto the ARM9 host.

3. THE ADOPTED STRATEGY

3.1 Platform and Tools

The MORPHEUS toolset flow described in Part III of the book has been employed for mapping the application onto MORPHEUS chip. Not all of the C files written for the application are processed by the MORPHEUS toolset flow. Test-data generator/checker routines have also been created alongside the files that describe the wireless components defined in section 1. These generator and checker routines are used for producing the data stream that is supplied to the MORPHEUS chip during the simulation phase. The Modelsim simulation environment is also provided with modules external to the MORPHEUS chip, required for managing the data-stream produced by the generator routines.

3.2 System overview

The wireless application was originally modelled as a 'C' program, running on a PC. The program, shown schematically in Figure 17-3, consists of a sample stream generator and file-writer function, the application mapping functions, applied to the data in sequence and finally a file-writer function, taking as input the bit-stream generated by the last application module.

The sample stream generator creates a data-stream consisting of 144-sample-point symbols that are forwarded to the Cyclic-Prefix Removal function. The same data-stream is formatted for dumping into a VHDL-readable ASCII file, which is used during the VHDL simulation phase (see Figure 17-5)

The bit-stream produced by the QAM Symbol Demapper function is dumped by the second file-writer into a VHDL-readable ASCII file. This second file is employed during the VHDL simulation phase for validating the output of MORPHEUS against the output produced by the reference 'C' simulation.

The "C" template is subsequently used as the starting point for application of the MORPHEUS toolset, which generate the configuration bitstreams loaded into MORPHEUS. The chosen configuration for the application is shown in Figure 17-4. Local SRAM is employed for double-buffering of sample data and of the generated bit-stream. The FFT resides in the DREAM array, and two DEBs are employed for communication with the rest of the application. The constellation demapper and guard-removal blocks are inside the FlexEOS array, again employing DEBs for I/O. A final buffer residing in the local SRAM is employed for aligning the generated

bit-stream to the external memory's word-length, in order to maximize transfer efficiency. All inter-block transfers are done by DMA channels, programmed and controlled by the ARM9 host.

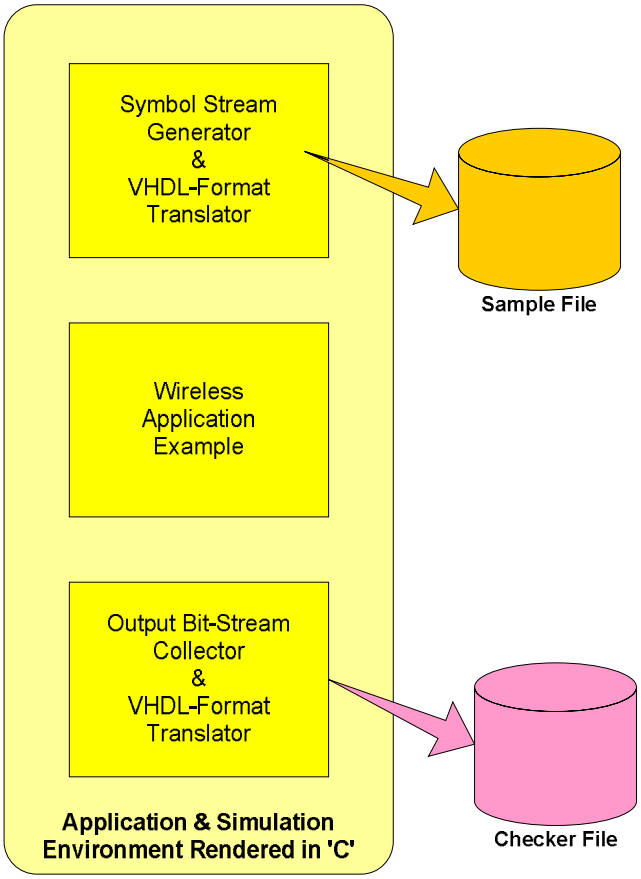

Figure 17-9. Outline of the application template/test-data generatorimplemented in 'C'

The executable binary produced by the MORPHEUS toolset is converted for use in a VHDL simulation environment, shown in Figure 17-5. This environment consists of the MORPHEUS modules representing the chip, attached via the external memory interface to a "ROM" model holding the executable image that is loaded and run by the ARM9 host of MORPEHUS chip. A data-stream driver is attached to the MORPHEUS GPIO, sending the translated patterns that were saved by the 'C' simulator data generator/file-writer. A bit-patterns checker is attached to the GPIO's output lines,

comparing the MORPHEUS output with the contents of the file that was generated by the C application bit-stream logger.

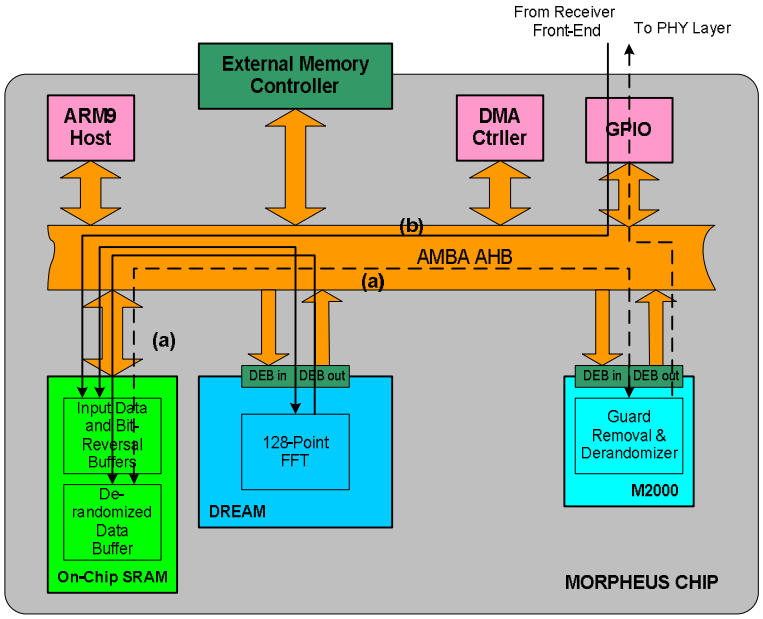

Figure 17-10. Illustration of the application mapping onto MORPHEUS

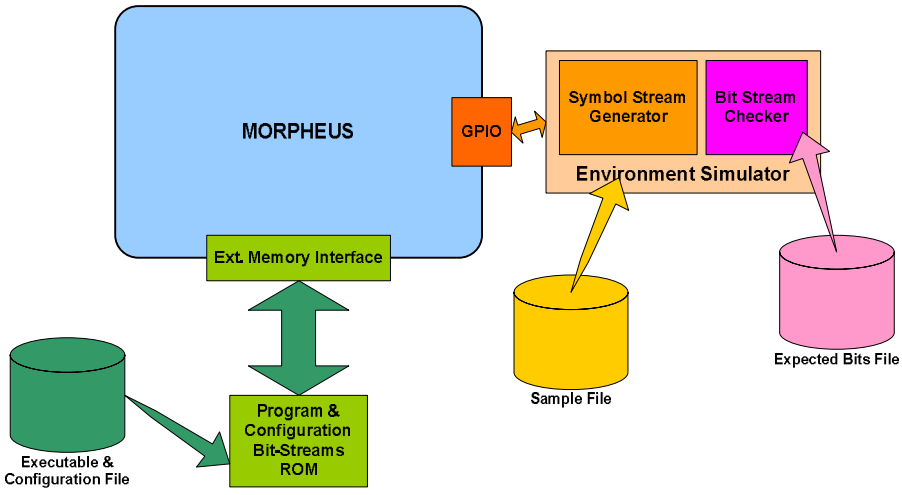

Figure 17-11. The Simulation environment for the wireless PHY segment on MORPHEUS

4. CONCLUSIONS

A small part of the PHY layer for the 802.16e/j wireless standard was ported onto the MORPHEUS architecture, a novel combination of conventional software-running processor with dynamically reconfigurable hardware accelerators. The porting was aided with the SPEAR and Molen Compiler tools that allow the capture of performance-critical code fragments and their replacement with library calls to an API that translates the passing of arguments to data-processing software into DMA transfers between processor-accessible memory and the hardware accelerators represented by the API primitives. Synchronization issues arising from the mapping of segments of sequential code onto concurrent hardware are also taken care of by the tools. An RTOS onto which the user application is attached manages the use of the hardware resources by the transformed application code.

A simulation of the resulting system in a ModelSim environment was employed to validate the porting process. Experience with the use of the platform and tools showed that the approach is promising. The tool-chain front-end is of particular importance from the user's point-of-view, and is seen as the most dynamic aspect in the area of tool development for platforms such as MORPHEUS. This is because most of the issues related to exploitation of concurrency and reconfigurability arise in the specification and capture phase. Thus, progress in this part of the tool chain can have a multiplicative effect on the scope and ease of use of platforms such as MORPHEUS.

REFERENCES

1. C.Eklund, et al., IEEE Standard 802.16: A Technical Overview of the WirelessMAN Air Interface for Broadband Wireless Access, IEEE Communications Magazine, June 2002, pp.98-107.
2. WiMAX Forum, "Mobile WiMAX - Part I: A Technical Overview and Performance Evaluation", Feb. 2006.
3. H.Yaghoobi, "Scalable OFDMA Physical Layer in IEEE 802.16 WirelessMAN", Intel Technology Journal, vol.8, no.3, August 2004, pp.201-212.
4. "IEEE Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems - Draft Amendment j: Multihop Relay Specification", IEEE P802.16j/D2, December 24, 2007.